IN THE SPECIFICATION

Please amend the specification as follows:

[25] FIG. 3 illustrates a method <u>1000</u>, in an embodiment, of clearing deferred load uops operable when an STD uop executes. When an STD uop executes, the scheduler may compare an identifier of the STD uop with dependency pointers of all uops stored by the scheduler (box 1110). If a match is detected, the scheduler may clear the dependency marker within a matching scheduler entry (boxes 1120, 1130). Thereafter, the method may terminate. When the marker of a previously deferred load uop is cleared, the load uops may be scheduled for execution according to the scheduler's normal processes.

[27] FIG. 4 is a block diagram of ~~an entry~~ <u>entries</u> 210<u>, 210.1, ..., 210.N</u> for a scheduler 200 when storing a load uop according to an embodiment of the present invention. The scheduler entry 210 may include fields 220, 230 for storage of the load uop itself and for storage of administrative information to be used by the scheduler during the execution of the load uop. For example, the admin field 230 may store one or more pointers 240, 250 (typically two) identifying data to be used to calculate a memory address from which data is to be loaded. According to an embodiment, the admin field 230 of a load uop may be extended to include fields for storage of a dependency pointer 260 and a valid flag 270. The dependency pointer 260 may identify the STD on which the load uop depends. The state of the valid flag 270 may be tested to determined whether or not scheduling of the load uop is to be deferred.